WEST

End of Result Set

Generate Collection

L5: Entry 1 of 1 File: TDBD Aug 1, 1978

TDB-ACC-NO: NN78081184

DISCLOSURE TITLE: Cursor Movement Control Circuitry. August 1978.

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, August 1978, US

VOLUME NUMBER: 21
ISSUE NUMBER: 3
PAGE NUMBER: 1184 - 1186

PUBLICATION-DATE: August 1, 1978 (19780801)

CROSS REFERENCE: 0018-8689-21-3-1184

DISCLOSURE TEXT:

3p. In data entry or data display systems it is desirable to be able to selectively position the cursor to a predetermined position on the display. In the past cursor movement was controlled by a set of keys which, upon being operated, would cause the cursor to move right, left, up or down. Generally this required repeated operation of a number of keys to effect the desired movement. A single switch, such as a digital joy stick, can be used to replace the keys and with appropriate circuitry to fly the cursor to any desired position on the display. - In Fig. 1, keyboard display 10 includes a display screen 11 upon which cursor 12 appears. Assume that it is desirable to move cursor 12 from position A to position B. Digital joy stick 20 would be operated to simultaneously engage contact elements 21 and 22 (Fig. 2), whereby cursor 12 would move up and to the right in a series of steps until it reaches position B. Contact 21 controls cursor movement in an up direction whereas contacts 22, 23 and 24 control movement in right, down and left directions, respectively. Contacts 21, 22, 23 and 24 are connected to inputs of AND circuits 57, 58, 59 and 60, respectively. These AND circuits are fed by Clock A, Clock B, Clock C and Clock D pulses, respectively. - The clock pulses are generated in a free-running manner. Consequently, the cursor 12 (Fig. 1) could move along one of two paths when moving from position A to position B, as shown, depending upon which clock pulse occurred first when joy stick 20 was operated. The clocks are generated as oscillator 30 (Fig. 2) provides pulses to clock generator 40. The clock pulses from clock generator 40 are applied as fast clock pulses directly to AND circuits 46, 49, 52 and 55 and as slow clock pulses via flip-flops 41, 42, 43 and 44 to AND circuits 45, 48, 51 and 54, respectively. When the cursor has to be moved over a relatively long distance, it is desirable to move it at a fast rate, but when it is to be moved only a short distance, then it travels at a slow rate. The slow rate AND circuits 45, 48, 51 and 54 are conditioned by the reset output of speed control latch 68, and the fast rate AND circuits 46, 49, 52 and 55 are conditioned by the set output of that latch. The slow and fast Clock A, Clock B, Clock C and Clock D AND circuits feed OR circuits 47, 50, 53 and 56, respectively. The outputs of these OR circuits provide the clock inputs for AND circuits 57, 58, 59 and 60, respectively. - The outputs of AND circuits 57, 58, 59 and 60 are applied to the set inputs of latches 61, 62, 63 and 64, respectively and to the inputs of OR circuit 65. OR circuit 65 controls the advancement of counter 66, which in turn controls the resetting of speed control latch 68. The outputs of latches 61, 62, 63 and 64 feed OR circuit 67 which in turn provides an input to AND circuit 70. AND circuit 70 is clocked by Clock E from clock generator 40. Clock E is also used to reset latches 61, 62, 63 and 64 via driver 69. Driver 69 provides sufficient delay to prevent any race condition. The output of AND circuit 70 resets counter 66 and speed control latch

68. Counter 66, upon reaching a predetermined count, sets speed control latch 68. - By the arrangment just described, when cursor 12 is in a static position, speed control latch 68 is reset, counter 66 is reset and a sequence of pulses from clock generator 40 is passed by the slow rate AND circuits 45, 48, 51 and 54 to the AND circuits 57, 58, 59 and 60 via OR circuits 47, 50, 53 and 56, respectively. The clock pulses, however, are not passed by AND circuits 57, 58, 59 and 60 until joy stick 20 is operated. When joy stick 20 is operated, then, depending upon which contact or contacts are made, the cursor will move in the appropriate direction, and its movement will be counted as pulses are passed by OR circuit 65 to counter 66. If joy stick 20 is still held in an operative position when counter 66 has reached the predetermined count, speed control latch 68 is set. Cursor 12 continues to move, but at the fast rate, until joy stick 20 is no longer held in an operative position. This is detected by OR circuit 67 when latches 61, 62, 63 and 64 are in the reset state. OR circuit 67 provides a positive input to AND circuit 70 when the latches just mentioned are in the reset state. Clock E is then passed by AND circuit 70 to reset speed control latch 68 so that, upon the next operation of joy stick 20, cursor 12 will begin its movement at slow speed.

:VALID *TDB-ACC-NO: NN78081184

DISCLOSURE TITLE: Cursor Movement Control Circuitry. August 1978.

PUBLICATION-DATA: IBM Technical Disclosure Bulletin, August 1978, US

VOLUME NUMBER: 21

ISSUE NUMBER: 3

PAGE NUMBER: 1184 - 1186

PUBLICATION-DATE: August 1, 1978 (19780801)

CROSS REFERENCE: 0018-8689-21-3-1184

DISCLOSURE TEXT:

3p. In data entry or data display systems it is desirable to be able to selectively position the cursor to a predetermined position on the display. In the past cursor movement was controlled by a set of keys which, upon being operated, would cause the cursor to move right, left, up or down. Generally this required repeated operation of a number of keys to effect the desired movement. A single switch, such as a digital joy stick, can be used to replace the keys and with appropriate circuitry to fly the cursor to any desired position on the display.

- In Fig. 1, keyboard display 10 includes a display screen 11 upon which cursor 12 appears. Assume that it is desirable to move cursor 12 from position A to position B. Digital joy stick 20 would be operated to simultaneously engage contact elements 21 and 22 (Fig. 2), whereby cursor 12 would move up and to the right in a series of steps until it reaches position B. Contact 21 controls cursor movement in an up direction whereas contacts 22, 23 and 24 control

movement in right, down and left directions, respectively. Contacts 21, 22, 23 and 24 are connected to inputs of AND circuits 57, 58, 59 and 60, respectively. These AND circuits are fed by Clock A, Clock B, Clock C and Clock D pulses, respectively.

- The clock pulses are generated in a free-running manner.

Consequently, the cursor 12 (Fig. 1) could move along one of two paths when moving from position A to position B, as shown, depending upon which clock pulse occurred first when joy stick 20 was operated.

The clocks are generated as oscillator 30 (Fig. 2) provides pulses to clock generator 40. The clock pulses from clock generator 40 are applied as fast clock pulses directly to AND circuits 46, 49, 52 and 55 and as slow clock pulses via flip-flops 41, 42, 43 and 44 to AND circuits 45, 48, 51 and 54, respectively. When the cursor has to be moved over a relatively long distance, it is desirable to move it at a fast rate, but when it is to be moved only a short distance, then it travels at a slow rate.

The slow rate AND circuits 45, 48, 51 and 54 are conditioned by the reset output of speed control latch 68, and the fast rate AND circuits 46, 49, 52 and 55 are conditioned by the set output of that latch. The slow and fast Clock A, Clock B, Clock C and Clock D AND circuits feed OR circuits 47, 50, 53 and 56, respectively. The outputs of these OR circuits provide the clock inputs for AND circuits 57, 58, 59 and 60, respectively.

- The outputs of AND circuits 57, 58, 59 and 60 are applied to the set inputs of latches 61, 62, 63 and 64, respectively and to the inputs of OR circuit 65. OR circuit 65 controls the advancement of counter 66, which in turn controls the resetting of speed control latch 68. The outputs of latches 61, 62, 63 and 64 feed OR circuit

67 which in turn provides an input to AND circuit 70. AND circuit 70 is clocked by Clock E from clock generator 40. Clock E is also used to reset latches 61, 62, 63 and 64 via driver 69. Driver 69 provides sufficient delay to prevent any race condition. The output of AND circuit 70 resets counter 66 and speed control latch 68. Counter 66, upon reaching a predetermined count, sets speed control latch 68.

- By the arrangment just described, when cursor 12 is in a static position, speed control latch 68 is reset, counter 66 is reset and a sequence of pulses from clock generator 40 is passed by the slow rate AND circuits 45, 48, 51 and 54 to the AND circuits 57, 58, 59 and 60 via OR circuits 47, 50, 53 and 56, respectively. The clock pulses, however, are not passed by AND circuits 57, 58, 59 and 60 until joy stick 20 is operated. When joy stick 20 is operated, then, depending upon which contact or contacts are made, the cursor will move in the appropriate direction, and its movement will be counted as pulses are passed by OR circuit 65 to counter 66. If joy stick 20 is still held in an operative position when counter 66 has reached the predetermined count, speed control latch 68 is set. Cursor 12 continues to move, but at the fast rate, until joy stick 20 is no longer held in an operative position. This is detected by OR circuit 67 when latches 61, 62, 63 and 64 are in the reset state. OR circuit 67 provides a positive input to AND circuit 70 when the latches just mentioned are in the reset state. Clock E is then passed by AND circuit 70 to reset speed control latch 68 so that, upon the next operation of joy stick 20, cursor 12 will begin its movement at slow speed.

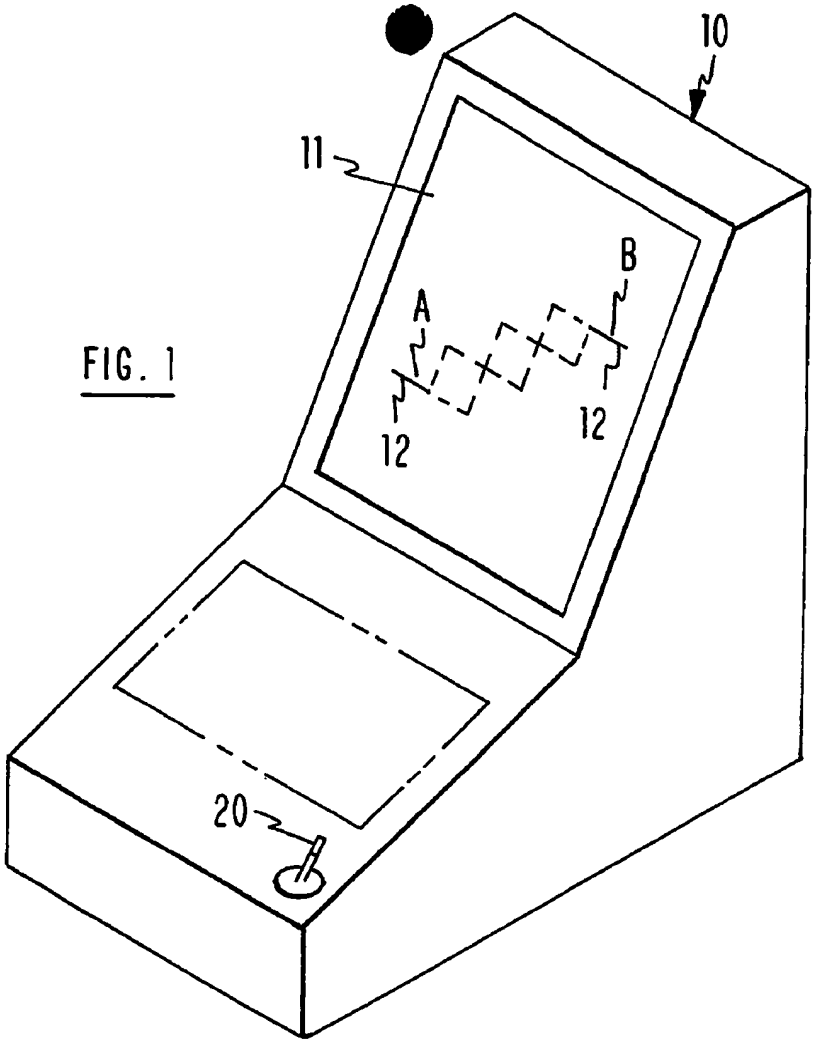

FIG. 1

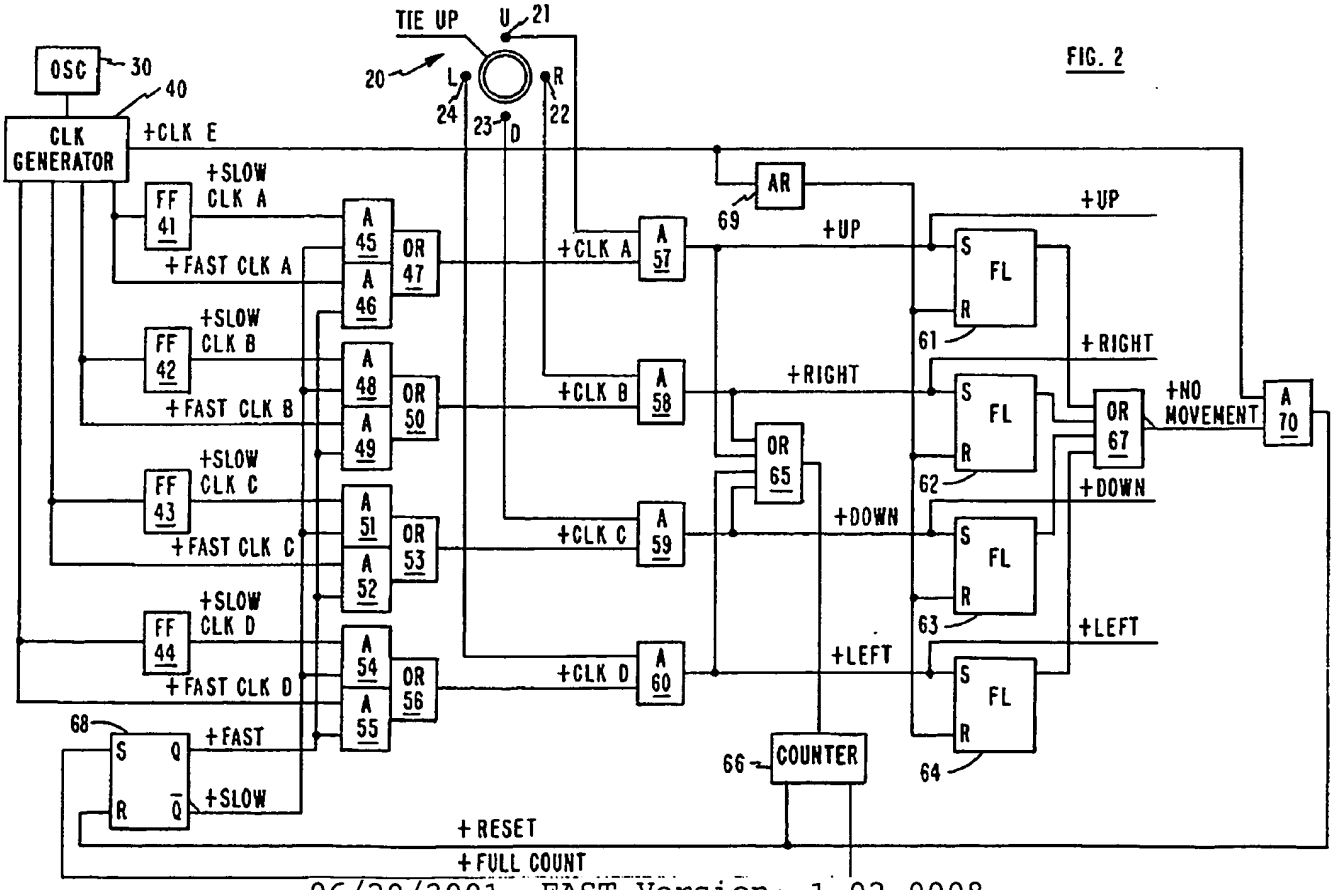

FIG. 2

WEST

Freeform Search

Database: US Patents Full-Text Database
US Pre-Grant Publication Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: 18 and 110

Display: 10 **Documents in Display Format:** CIT **Starting with Number** 1

Generate: ○ Hit List ◉ Hit Count ○ Image

Search | Clear | Help | Logout | Interrupt

Main Menu | Show S Numbers | Edit S Numbers | Preferences

Search History

Today's Date: 6/20/2001

DB Name	Query	Hit Count	Set Name
USPT	l8 and l10	9	L11
USPT	((345/342)!.CCLS.)	250	L10
USPT	345/342.ccls	0	L9
USPT	l7 and @ad<19980904	306	L8
USPT	l4 and l5	323	L7
USPT	l3 and l5	1	L6
USPT	(348/$)!.CCLS. or 345/$.ccls.	63624	L5
USPT	window same (accelerat$ or speed)same move$	2351	L4
USPT	pip same (accelerat$ or speed)same move$	18	L3
USPT	pip same (accelerat$ or speed)	266	L2
USPT	window same (accelerat$ or speed)	13652	L1

09/389,557

WEST

Generate Collection

L20: Entry 3 of 32 File: USPT Jul 11, 2000

DOCUMENT-IDENTIFIER: US 6088023 A
TITLE: Integrated pointing and drawing graphics system for computers

BSPR:
In the preferred embodiment, the presenting steps include displaying a window having the same shape as the input surface. Also, the method steps further preferably include relocating the window when in the relative mode and relocating the cursor when in the absolute mode. In the presently preferred implementation, the window defines a periphery and the input surface defines a periphery, and the moving step includes defining a border within the periphery of the input surface. Also, the moving step includes presenting an image of the border within the periphery of the window. Per the present invention, the window is caused to move when the contact element stimulates the input surface between the periphery of the input surface and the border. Still further, the moving step includes moving the window when the contact element moves over the input surface at a speed greater than a predetermined speed. If desired, the window can be enlarged or shrunk in size.

WEST

Freeform Search

Database: US Patents Full-Text Database
US Pre-Grant Publication Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: window near3 (position or move or reposition) same (video or image) same (accelerat$ or speed)

Display: 10 **Documents in Display Format:** CIT **Starting with Number** 1

Generate: ○ Hit List ◉ Hit Count ○ Image

Search | Clear | Help | Logout | Interrupt

Main Menu | Show S Numbers | Edit S Numbers | Preferences

Search History

Today's Date: 6/20/2001

DB Name	Query	Hit Count	Set Name
USPT	window near3 (position or move or reposition)same (video or image) same ((accelerat$ or speed)near3 (mouse or joystick or cursor or pointer))	1	L23
USPT	window near3 (position or move or reposition)same (video or image) same (accelerat$ or speed)	93	L22
USPT	window near3 (position or move or reposition)	17382	L21
USPT	window same (video or image) same (accelerat$ or speed)same (position or move or reposition) same (mouse or joystick or cursor or pointer)	32	L20
USPT	window same (video or image) same (accelerat$ or speed)same (position or move or reposition)	479	L19
USPT	window same (video or image) same (accelerat$ or speed)	1865	L18
TDBD	(picture adj picture) same (video or image or sprite) same (accelerat$ or speed)	0	L17
DWPI	(picture adj picture) same (video or image or sprite) same (accelerat$ or speed)	9	L16
EPAB	(picture adj picture) same (video or image or sprite) same (accelerat$ or speed)	4	L15
JPAB	(picture adj picture) same (video or image or sprite) same (accelerat$ or speed)	30	L14
PGPB	(picture adj picture) same (video or image or sprite) same (accelerat$ or speed)	0	L13
USPT	"picture in picture"	0	L12
USPT	l10 not l1	63	L11
USPT	(picture adj picture) same (video or image or sprite) same (accelerat$ or speed)	65	L10
USPT	(picture adj picture)	2603	L9
USPT	(picture adj in adj picture)	0	L8
USPT	(picture adj in adj picture) same (video or image or sprite) same (accelerat$ or speed)	0	L7
TDBD	pip same (video or image or sprite) same (accelerat$ or speed)	1	L6
DWPI	pip same (video or image or sprite) same (accelerat$ or speed)	4	L5
EPAB	pip same (video or image or sprite) same (accelerat$ or speed)	2	L4
JPAB	pip same (video or image or sprite) same (accelerat$ or speed)	0	L3
PGPB	pip same (video or image or sprite) same (accelerat$ or speed)	0	L2
USPT	pip same (video or image or sprite) same (accelerat$ or speed)	41	L1

09/389,557

WEST

Generate Collection

L22: Entry 45 of 93 File: USPT Aug 15, 1995

DOCUMENT-IDENTIFIER: US 5442390 A
TITLE: Video on demand with memory accessing and or like functions

DEPR:
In the preferred embodiment of the invention, as shown in FIG. 6, the video 100 is transferred as a moving or rolling viewing window called a video segment 200 having a time-interval based span or size. The video 100 is transferred by moving the viewing window or video segment 200 forward, from the start to the end, at a substantially constant speed to coincide with the play-back speed of the video 100. However, in response to the customer commands entered on the video controller 13 the position of the moving viewing window can be altered.

WEST

Generate Collection

L22: Entry 48 of 93 File: USPT May 9, 1995

DOCUMENT-IDENTIFIER: US 5414455 A
TITLE: Segmented video on demand system

DEPR:
The video 100 is transferred by moving the viewing window or video segment 200 forward, from the start to the end, at a substantially constant speed to coincide with the play-back speed of the video 100. However, in response to the customer commands entered on the video controller 13 the position of the moving viewing window can be altered.

L22: Entry 66 of 93 File: USPT Sep 25, 1990

DOCUMENT-IDENTIFIER: US 4959803 A
TITLE: Display control system

BSPR:
The disadvantage of the system of FIG. 5(b) involving a mapping table is as follows. The addresses of the image data in the window memory 4 are output sequentially from the mapping table 5 so that the image data stored at the addresses are read in shared time and displayed directly on the CRT 6. With this system, it is possible to move the window quickly. On the other hand, however, since graphic drawing in the window memory 4 is also performed in shared time, drawing speed is slow.

BSPR:
The disadvantage of the system of FIG. 5(c) involving clipping is as follows. The code data of unnecessary image data in the segment buffer 7 is removed by clipping before the code data of the image data stored in the segment buffer 7 is transferred for graphic drawing on the display memory 8 and displayed on the CRT 9. To move the window on the CRT 9, therefore, the system requires hardware for drawing graphics at a high speed on the display memory 8 from the code data stored in the segment buffer 7.

BSPR:
Accordingly, in view of the prior art technology described above, the object of the present invention is to provide an image display control system which incorporates the advantages of a system using raster operation and the advantages of a system using a mapping table, so that in the display mode it is possible to move the window on the display quickly while watching the screen and, in the graphic drawing mode it is possible to draw and edit graphics rapidly in the window memory, thus allowing the operator to edit a document at a high speed while watching the CRT screen.

DEPR:
According to the present invention, as understood from the above, the display control system comprises a display memory having a random access port and a serial access port, a window buffer memory for storing image data, a window controller for controlling the position of the window buffer memory content displayed on a display device, and a selection circuit for setting the operation mode of the window buffer memory to the display mode or to the graphic drawing mode. When the display mode is selected, the window buffer memory content is presented directly on the display device in shared time as the position of the data displayed is controlled by the window controller. When the graphic drawing mode is selected, image data transmission between the window buffer memory and the display memory or graphic drawing in the window buffer memory is conducted without time sharing. Consequently, it is possible for an operator to move a window at a high speed while watching the picture on the CRT when the selection circuit is switched over to the display mode, and to draw and edit graphics in the window buffer memory at a high speed when the selection circuit is switched over to the graphic drawing mode. As a result, it is possible to edit a document acitvely and rapidly while watching the picture on the display device.